Cost Efficient Distributed Load Frequency Control in Power Systems

Flavio R. de A. F. Mello, Dimitra Apostolopoulou, Eduardo Alonso

City, University of London
London, UK EC1V 0HB
Email: {flavio.ribeiro-de-aquino-f-mello, dimitra.apostolopoulou, e.alonso}@city.ac.uk

Abstract: The introduction of new technologies and increased penetration of renewable resources is altering the power distribution landscape which now includes a larger numbers of micro-generators. The centralized strategies currently employed for performing frequency control in a cost efficient way need to be revisited and decentralized to conform with the increase of distributed generation in the grid. In this paper, the use of Multi-Agent and Multi-Objective Reinforcement Learning techniques to train models to perform cost efficient frequency control through decentralized decision making is proposed. More specifically, we cast the frequency control problem as a Markov Decision Process and propose the use of reward composition and action composition multi-objective techniques and compare the results between the two. Reward composition is achieved by increasing the dimensionality of the reward function, while action composition is achieved through linear combination of actions produced by multiple single objective models. The proposed framework is validated through comparing the observed dynamics with the acceptable limits enforced in the industry and the cost optimal setups.

Keywords: Multi-Agent Reinforcement Learning, Multi-Objective Reinforcement Learning, Frequency Control, Economic Dispatch, Deep Deterministic Policy Gradient

1. INTRODUCTION

Over recent years, the field of electrical power systems has been experiencing the beginning of what may prove to be a structural transformation. Renewable resources have been increasing their penetration in the marketplace, which may displace traditional sources. Decreasing costs of solar panels lead to increased adoption in households, to the extent that there are already legal provisions for household customers to sell stored energy back into the electrical grid as mentioned in Ambrose (2019). Vehicle to grid and smart charging technologies are posed to enable electric cars to contribute to balancing the power grid see, e.g., Steitz (2019), and Ali et al. (2017). In Leggett (2017) it is mentioned that by 2030 there will be nine million electric vehicles that need to be charged by National Grid transmission system. The aforementioned event will lead to an increment in the diversity and quantity of sources which are able to inject power into the electrical grid. This represents a significant increase in the complexity of the grid, shifting away from a small number of large scale producers to include an ever increasing number of micro-sized sources in the form of individual households, electric cars, etc. Such manifold structure, in turn, will intensify the need for intelligent, automated and decentralized control solutions.

Modern electrical energy distribution is largely done by means of wide-ranging synchronous grids. Being synchronous means the entirety of the grid is electrically connected and thus every element attached to the grid share the same observed operating frequency. This is true for both the consumers as well as the producers (generators). In these systems the observed operating frequency changes over time according to i) the total power being injected into the system by all the generators; ii) the total power being consumed by all loads. To electrically balance the system, independent system operators (ISOs) send signals to generators to modify their output such that load and generation are balanced and the system frequency is nominal.

The primary objective of this paper is to investigate the feasibility of leveraging Reinforcement Learning (RL) techniques for training autonomous, decentralized agents able to perform frequency control in an electric power system according to two distinct hierarchical objectives: i) maintain the system frequency within predefined tolerated limits; ii) minimize the cost of production.

Multiple techniques have already been proposed to achieve frequency control decentralization. From a traditional control standpoint, Apostolopoulou, et al propose methods for approximating the automatic generation control (AGC) algorithm while solving the economic dispatch in semi-decentralized fashion by restricting the Balancing Authority (BA) areas communication and, thus, avoiding congestion associated with the exponential increase of connections in the network (see Apostolopoulou et al. (2015a) and Apostolopoulou et al. (2015b)). Additionally, Model Predictive Control (MPC) techniques have been proposed to perform decentralized frequency control whilst satisfying predetermined constraints (see Ali et al. (2017), Kumtepeli et al. (2016) and Heydari et al. (2019)). In the

Reinforcement Learning realm, Rozada (2018) proposes the use of Multi-Agent Reinforcement Learning (MARL) techniques, more specifically, the MADDPG algorithm, which proved able to successfully perform primary and secondary control but failed to perform tertiary control. Despite being separate layers of control, both primary and secondary control share a common overarching objective related to frequency deviation. Tertiary control, however, is associated with a slightly different objective: to minimize the total cost of electricity production. These differences in objective alignment could explain why the MAADPG algorithm, as implemented in said paper, successfully performed primary and secondary controls but failed with tertiary control. For this end, this paper proposes the addition of MORL techniques to the algorithm.

In this paper we i) frame the frequency control problem as a Markov Decision Process to allow for the use of reinforcement learning techniques (Section 2); ii) propose the incorporation of two distinct multi-objective reinforcement learning techniques to the MADDPG algorithm to perform frequency control in a cost-efficient way (Section 3); iii) compare the performance of both techniques through numerical studies (Section 4); and iv) draw conclusions from the observed behaviours (Section 5).

2. BACKGROUND

In this section, the frequency control problem is formulated and the reinforcement learning techniques employed to perform such control are presented.

2.1 Load frequency control and Economic Dispatch

Frequency control can be divided into three hierarchical layers: Primary, Secondary and Tertiary control.

Primary control acts to counterbalance changes in the total system load by adjusting the output levels of all generators attached to the grid by an amount proportional to the difference between the observed and nominal frequency. Droop Control would be the most commonly applied form of primary control, see Miller and Malinowski (1994). Primary control has the benefit of being completely decentralized as each generator is able to observe individually the current frequency in the system. However, this technique has its limitations as it results in steady state errors and ignores any economical implications.

Secondary control aims to further balance the power grid by acting upon the steady state error resulted by the limitations in primary control. To this end, Automatic Generation Control (AGC) algorithms are often employed, see Miller and Malinowski (1994). However, these algorithms are often centralized to some extent, with an individual entity overseeing the entire grid and issuing commands for the individual generators. When performing secondary control, the following set of equations apply to the frequency deviation problem:

$$P(t+1) = P(t) + \frac{Z_{\text{total}}(t) - \frac{1}{R_D}\Delta\omega(t) - P(t)}{T_G}, \quad (1)$$

$$\omega(t+1) = \omega(t) + \frac{P(t+1) - L(t) - D \cdot \Delta\omega(t)}{M}, \quad (2)$$

$$Z_{\text{total}}(t) = \sum_{i=1}^{I} Z_i(t), \quad (3)$$

$$L_{\text{total}}(t) = \sum_{j=1}^{J} L_j(t), \quad (4)$$

$$\Delta\omega(t) = \omega(t) - \omega_{\text{nominal}}, \quad (5)$$

where $P(t)$ is the total power injected into the grid at time t, $Z(t)$ is the secondary control action, $Z_i(t)$ is the control action of generator i at time t, $L(t)$ is the load at time t, $L_j(t)$ is the load by consumer j at time t, $\omega(t)$ is the system frequency at time t, R_D is the droop control coefficient selected for the system, D is the damping coefficient of the system, M is the electrical inertia of the grid, I is the number of generators, J is the number of loads in the system, and ω_{nominal} is the nominal frequency.

Also referred to as Economic Dispatch, the objective of tertiary control is to minimize the total production costs of the grid. Doing so requires a centralized entity with knowledge of each generating power output and cost of production curve, as well as relevant physical limits with regards to minimum and maximum output levels. For estimating the cost associated with reaching a given power output, a typically quadratic or piecewise linear function is used to express the operating cost as a function of power output as shown below:

$$C_i(t) = \alpha_i + \beta_i p_i(t) + \gamma_i p_i^2(t), \text{ for } i = 1, \ldots, I, \quad (6)$$

where for generating unit i at time t, $p_i(t)$ is the power output, $C_i(t)$ is the cost of production, and α_i, β_i, γ_i are constants.

Traditionally, a multitude of iterative techniques may be employed for dealing with Economic Dispatch, as seen in Wood et al. (2013). All such techniques require a centralized controller to calculate the optimal setup and issue commands to every generator in the network. In a scenario with increasing numbers of generators attached to the grid, the centralized approach to Economic Dispatch becomes ever more complex due to the increase in computational power necessary to find the optimal arrangements. For a power system with I generators and J loads, the following set of equations apply at time t:

$$\begin{aligned} &\underset{p_i(t)}{\text{minimize}}\ C_{\text{total}}(t) = \sum_{i=1}^{I} C_i(t), \\ &\text{such that } \sum_{i=1}^{I} p_i(t) = \sum_{j=1}^{J} L_j(t), \\ &\qquad p_i^{min} \le p_i(t) \le p_i^{max}, \text{ for } i = 1, \ldots, I, \end{aligned} \quad (7)$$

where $C_{\text{total}}(t)$ is the total cost of production at time t, $C_i(t)$ is the cost of production of generator i at time t, $p_i(t)$ is the output level of generator i at time t, p_i^{min} and p_i^{max} are, respectively, the minimum and maximum output levels for generator i, and $L_j(t)$ is the power consumption of load j at time t. Solving tertiary control entails finding the power output combination set $\{p_1^\star(t), p_2^\star(t), p_3^\star(t), \ldots, p_I^\star(t)\}$ that minimizes the global

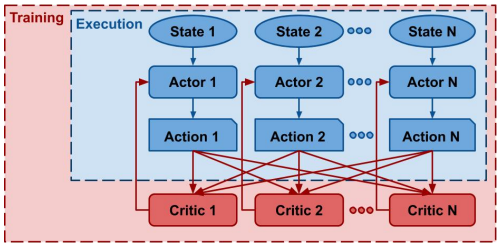

Fig. 1. Actor/Critic relationship in MADDPG

cost $C_{\text{total}}(t)$ while respecting the constraints of keeping the system balanced and every generator output within its operational limits, i.e., satisfying the constraints of (7). As seen in (1)-(5), the system reaching steady state operation entails that $\Delta\omega(t) = 0$ and, therefore, $P(t) = Z_{\text{total}}(t) = \sum_{i=1}^{I} p_i^{\star}(t)$.

2.2 Reinforcement Learning

Reinforcement Learning can be defined as a family of techniques used to train agents based on their interactions with the environment and the associated rewards/punishments observed. Given enough observations the trained agent becomes able to issue commands so as to find an optimal policy. The problem approached in this paper can be classified as fully cooperative as the agents work together to reach two objectives: i) electrically balance the system within the tolerated range indicated by the frequency deviation; ii) minimize the total cost of production. Multi-Objective Reinforcement Learning (MORL) relates to RL problems with multiple, sometimes conflicting, objectives. Successfully trained MORL agents should be able to perform tradeoffs, intentionally sacrificing adherence to one objective while advancing towards a more desired global state. To this end, there are a number of different techniques that can be employed, ranging from weighted-sum to Pareto dominating policies see, e.g., Liu et al. (2015), and Moffaert and Nowé (2014). The choice of which approach to take becomes an integral part of the design process of the solution.

The technique used in this paper is named Multi-Agent Deep Deterministic Policy Gradient (MADDPG) and is considered an extension of Deep Deterministic Policy Gradient (DDPG), combined with some elements of actor-critic RL techniques see, e.g., Lowe et al. (2017a), Lowe et al. (2017b). The MADDPG algorithm applies the actor-critic concept to multi-agent scenarios by centralizing learning whilst decentralizing execution, see Fig. 1. Once trained, the agents rely solely on their actors to take actions in the execution environment. Actors, therefore, remain decentralized in nature, having access only to the same information said agent would have in execution time. The critics, however, are centralized and have additional information in the form of the actions taken by all the other actors in the system.

3. PROPOSED FRAMEWORK

In this section, the details of the proposed implementation are described. This includes the neural networks architectures, the guidelines used for determining the reward functions used, and the approaches taken to incorporate multi-objective capabilities in the trained agents.

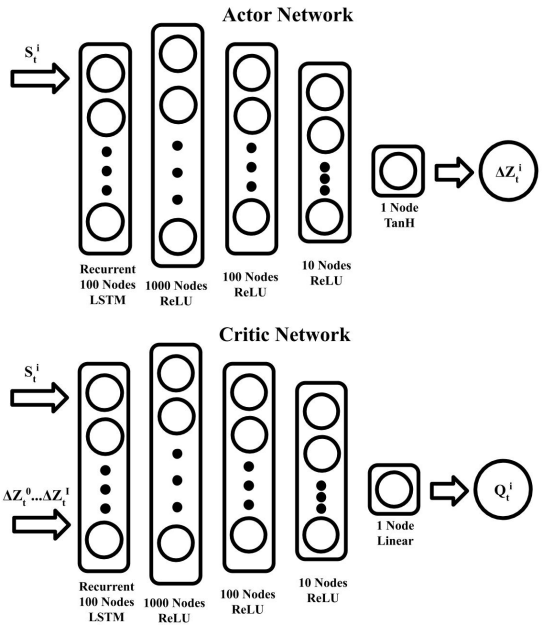

Fig. 2. Actor and Critic neural networks, where LSTM refers to long short-term memory cells and ReLU refers to the rectified linear unit

3.1 Neural Networks

The MADDPG algorithm leverages fully connected deep neural networks to model both the actor and the critic. In this study, both networks follow the same schema, with slight changes in the input/output layers. Additionally, this study employs the same algorithm to learn different policies to achieve different objectives. Often this requires changes in both the reward function and the set of variables that compose the $S_i(t)$ input, i.e., the state observed by agent i at time t. These changes are further described in Section 4 on a case by case basis. Common among all case studies are the output layers. The actor network outputs the action, in the form of change in total secondary action $(\Delta Z_i(t))$, where $Z_i(t) = \Delta Z_i(t) + Z_i(t-1)$, to be taken by its respective generator at time t. The critic network takes as input the outputs from all actor networks $(\Delta Z_0(t), \ldots, \Delta Z_I(t))$ and outputs the estimated quality, in the form of a Q-value, for that state-action for its respective generator. The base neural networks used are depicted in Fig. 2.

3.2 Reward Function Design

The reward function plays a pivotal role in the success of a Reinforcement Learning model. In multi-objective scenarios, the proportion between each reward component has increased importance. With these characteristics in mind, a collection of guiding principles shaped the design process of the reward functions used, namely: Finite upper and lower bounds act as points of reference for comparing given rewards, thus becoming easier to assess their quality. Individual reward functions are determined for each objective and the global reward is a composition of all individual functions. Such compositions are usually done by either

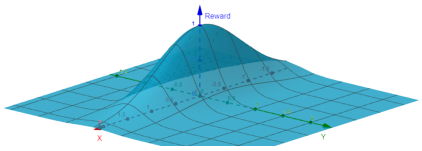

Fig. 3. Plot of a sample multi-objective reward function $r(x, y) = 2^{-x^2} \cdot 2^{-10 \cdot (y^2)}$

multiplication or addition. While keeping the adherence to all other objectives constant, increasing adherence to a given objective should monotonically increase the total reward. This is only possible if the objectives are not intrinsically contradictory. Having the global maxima of all individual objectives reward functions coincide means that the state which provides the maximum reward globally is the same which maximizes rewards for all individual objectives. For the purpose of streamlining the design process of the reward functions, all individual reward functions share the same base function:

$$f(x) = a \cdot 2^{-b \cdot x^2}, \tag{8}$$

where x is the input of the reward function, which varies according to the objective (e.g. $\Delta\omega$ for balancing frequency), and a and b are parameters in $\mathbb{R}^+$. This function provides some useful traits: It is symmetric with respect to the y-axis, which is instrumental if the objective is to minimize deviation. Besides, the base function has one single maximum at the origin, which means that composition by either multiplication or addition retains a single global maximum at the same point. Finally, parameters a and b can be used ad hoc for deforming the function while keeping the symmetry and maximum location characteristics.

3.3 Multi-Objective

This investigation sets out to test two distinct strategies for obtaining multi-objective optimization: reward-composition and action-composition. The former strives to accomplish the overall objective by learning a single policy that is able to fulfill multiple objectives. This is achieved by consolidating multiple objectives and their hierarchical relationship into a single reward function. An example of a multi-objective reward function can be seen in Fig. 3. Conversely, the action-composition approach trains one single-purpose set of agents per objective. During execution, actions from all sets of agents are consolidated into individual final actions. For a system with K agents and M objectives this composition is expressed as:

$$A_k(t) = \sum_{m=1}^{M} \rho_m \cdot \tilde{A}_k^m(t), \tag{9}$$

$$\sum_{m=1}^{M} \rho_m = 1, 0 \leq \rho_m \leq 1, \tag{10}$$

where $A_k(t)$ is the action to be taken by agent k at time t, ρ_m is the weight given to objective m, $\tilde{A}_k^m(t)$ is the action assigned to agent k, at time t, by the model aimed at optimizing objective m.

When performing action-composition, the reward functions used for each overarching objective does not intrinsically carry information regarding such preferences, these are declared in the form of the weights ρ_m, for $m = 1, ..., M$ used in runtime. One prerequisite for performing action composition is for the action-space to be quantitative. In categorical action environments, action consolidation cannot be done via arithmetic operations.

3.4 Reward Composition vs Action Composition

We propose two different methods of achieving multi-objective learning, reward composition and action composition. Besides observed performance, there are multiple factors that are taken into account when choosing a technique to be used in an industrial setting. In that sense, it can be argued that the action composition approach is superior from a systems design standpoint. Among the benefits provided by this strategy, one can single out the following: Breaking down the global model into a single objective ones decreases coupling between the models, facilitates reuse, and simplifies debugging (Separation of Concerns). Crafting bespoke multi-objective reward functions is a time-consuming enterprise. Breaking down into single objective rewards could speed up development as single objective reward functions behave in a more predictable way (Simplified Modeling). Declaring the objective priorities at the runtime means that these priorities can be seamlessly changed. Furthermore, finding the optimal priorities ratio can be done faster as the test feedback loop is tighter (Variable Priorities). Individual models can have different inputs. If different objectives of the system are associated with different Service Level Agreements (SLAs), the information sources which provide the inputs can be designed to match these SLAs. In a single model, all inputs are necessary to sample the actions, therefore have to provide an SLA that is compatible with the most critical objective. Using the studied scenario as an example, balancing the system frequency is critical at all times while optimizing for cost albeit still important is something that can be overlooked in critical situations. If those objectives are tackled by individual models, the inputs for balancing the system should be kept available and with minimum delay at all times. Conversely, the inputs for optimizing the cost can have their requirements relaxed — if they become offline, the system still can be operated at a degraded level by relying only on the frequency balancing model (Separate Data Sources).

3.5 Decentralization

We are using Multi-Objective RL techniques to solve primary, secondary, and tertiary control in a multi-agent-based model. The system designed in this analysis, albeit decentralized from the decision-making standpoint, still relies on some centralized information regarding the current state of the system, in particular $Z_{\text{total}}(t)$ the secondary control action at each time t. Although not completely fulfilling the decentralization requirement, this marks an important step towards full decentralization, as it changes the nature of the centralized entity from a fully-fledged decision maker to an information broker.

4. NUMERICAL STUDIES

The software developed for performing these case studies is fully configurable and allows for further experimentation with different configurations for electrical systems with any number of loads and generators electrical constants, and even reward functions and state inputs. The source code is open for future use and can be found at `https://github.com/melloflavio/2019-MSc_Thesis`.

4.1 Electrical System

We performed a multitude of experiments aimed at assessing the feasibility of leveraging multi-objective techniques to perform primary, secondary and tertiary control in an electrical power system. In order to perform the control experimentation, an electrical system simulator was implemented according to the equations described in Section 2.1. A consistent system topology was used across all experiments: three generators (G1, G2, and G3) and one single load (L1). The electrical constants were also kept the same for all the experiments and are shown in Table 1. In this context, pu refers to the 100 MVA base power used throughout this paper.

Table 1. Electrical System Constants

Term	Name	Value
R_D	Droop coefficient	0.1 pu
T_G	Time constant	30 s
d	Damping coefficient	0.0160 pu
M	Electrical inertia	0.1 pu

Being frequency control a continuous matter, each simulation begins at $t = 0$ considering that the system is fully balanced ($P(0) = L(0)$, $\Delta\omega(0) = 0$) and a perturbation occurs at t_0 in the form of a change in the total load. The task being performed then is to balance the system after this initial perturbation. In the interest of increasing the robustness of the models trained, the application developed is able to introduce noise in the simulated environment in the form of changing the initial values for the loads and generators power levels. The noise takes the form of a uniform distribution with magnitude of 0.5% of the initial value.

For each generator, a distinct cost profile was selected with the purpose of ensuring that the optimal setup is such that no generator is in either minimum (0.5 pu) or maximum (3.0 pu) output values. This helps evaluate the ensuing results as successful models should be able to steady the outputs around given values, rather than relying on the enforcement of minimum/maximum limits. Table 2 indicates the cost profiles of all generators:

Table 2. Generator Cost Profiles

Generator	α [\$/h]	β [\$/(h · MVA)]	γ [\$/(h · MVA^2)]
G1	510.0	7.7	0.00142
G2	310.0	7.85	0.00194
G3	78.0	7.55	0.00482

4.2 Case Study I - Frequency Control

The first experiment was ran to minimize the frequency deviation from nominal on an initially unbalanced electrical system. The only state input is the frequency deviation

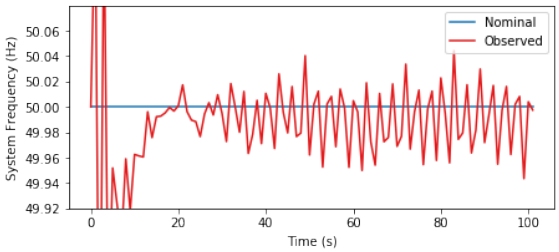

Fig. 4. Case I: Observed frequency

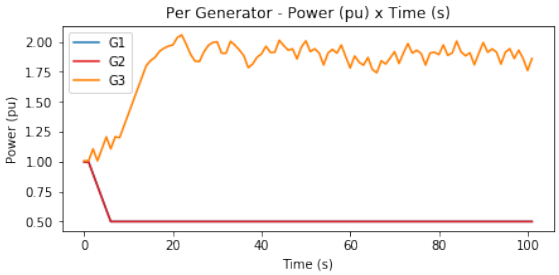

Fig. 5. Case I: Generator output

from the nominal setpoint ($\Delta\omega_t$) at each time t as defined in (5). The reward function, shown in (11), conforms to the guidelines set in Section 3.2 and is defined as follows:

$$r_I(\Delta\omega_t) = \left(9 \cdot 2^{-\frac{\Delta\omega_t^2}{2}} + 2^{-\frac{\Delta\omega_t^2}{100}}\right) \frac{1}{10}. \tag{11}$$

The results are depicted in Figs 4 and 5. After approximately 20 seconds, the load was successfully balanced and the power output and system frequency oscillates within 0.05 Hz (0.1%) of the nominal setpoint, which falls inside the accepted range of 0.5 Hz (1%) established by National Grid Electricity Transmission (2017). Overall, the results indicate that the implementation of the MADDPG algorithm works as expected.

In this case study we may see that the learned strategy is to have two generators reach their minimum output as fast as possible, while the third generator controls its output to stabilize the system gradually reducing the steady-state error. This cooperation by omission approach does not appear to be the most efficient way to balance the system. One possible reason for this behaviour could be that reaching the maximum/minimum limits may be the best way to ensure stable output for the other generators, as these limits are enforced in the simulation, and not in the modelled neural networks themselves (i.e., once the secondary action reaches whichever limit, the neural network may still issue commands to go beyond such limits, but they are disregarded by the electrical system simulation). Perhaps training with more diverse loads that better cover the full spectrum of the systems total power capacity would lead to more complex and robust cooperative strategies. On a further note regarding this cooperation by omission, it should be observed that the generator which is elected to effectively perform the balancing seems to be arbitrary. Rerunning the exact same experiment multiple times results in different generators being elected to perform this role. This is expected as all generators are identical with respect to their output

capabilities. While the impact of this choice is nonexistent for frequency control, this characteristic has repercussions when this model is combined with a cost optimization one to perform action composition, as will be shown in case study III.

4.3 Case Study II - Reward Composition: Cost and frequency deviation minimization

This experiment follows the reward-composition strategy described in Section 3.3. To this end, the state used as input in the algorithm is a triplet containing $\Delta\omega(t)$, $Z_i(t)$ and $Z_{\text{total}}(t)$. Additionally, a single reward function that reflects both objectives was crafted following the guidelines set in Section 3.2 and may be written as follows:

$$r_{II}(\Delta P_{\text{total}}(t), \Delta\omega(t)) = f(\Delta P_{\text{total}}(t)) \cdot g(\Delta\omega(t)), \quad (12)$$

$$f(\Delta P_{\text{total}}(t)) = 2^{\frac{-\Delta P^2_{\text{total}}(t)}{4}}, \quad (13)$$

$$g(\Delta\omega(t)) = \left(9 \cdot 2^{-\frac{\Delta\omega^2(t)}{2}} + 2^{-\frac{\Delta\omega^2(t)}{100}}\right)\frac{1}{10}. \quad (14)$$

The frequency component — $g(\Delta\omega(t))$ — is similar to the function used in case study I. The cost component — $f(\Delta P_{\text{total}}(t))$ — is expressed in terms of the total power deviation from the cost optimal setup with a normalization component denoted by

$$\Delta P_{\text{total}}(t) = \sum_{i=1}^{I} \left| \frac{p_i(t)}{p_i^\star(t)} - 1 \right|, \quad (15)$$

where $p_i(t)$ is the power produced by generator i at time t, and $p_i^\star(t)$ is the power output of generator i at time t which minimizes the total cost for the total output of all generators observed at time t (i.e., the output for generator i which is the solution to the minimization problem described in (7)).

Defining the cost component in terms of the deviation from optimal was performed with the intent of breaking an indirect relationship between both goals. For decreasing the total cost of production, there are two possible methods: 1) Change the operating output of all generators to achieve a lower cost of production while keeping the same total output — this keeps the system balanced and can be done until the optimal setup is reached. 2) Lower the total output — this can be done indefinitely at the cost of breaking the electrical balance. The reward function should remove the option of increasing the earned reward by simply lowering the total output.

The results of the experiment can be seen in Figs 6 and 7. Generators G1 and G2 follow closely their optimal outputs for the given total output at any given point. While G3 moves directly to and remains at the minimum output. Such behaviour could be interpreted as being associated with G3's optimal output being close enough to the minimum value that the model as a whole benefits more by having G3 remain at a flat level, and thus providing more certainty to G1 and G2, than by actively attempting to follow its optimal value.

Regarding frequency control, it still performs worse than the single-objective model seen in case study I. In this

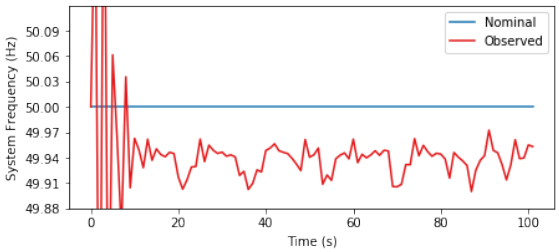

Fig. 6. Case II: Observed frequency

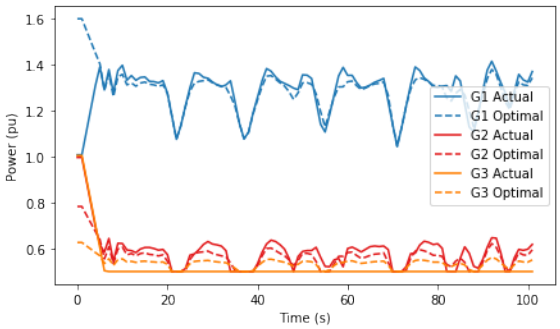

Fig. 7. Case II - Generator output vs cost optimal

case, the frequency remains within 0.12 Hz the nominal value, and exhibits a consistent downward shift of approximately 0.05 Hz. This falls inside the the accepted range of 0.5 Hz (1%) established by National Grid Electricity Transmission (2017).

4.4 Case Study III - Action Composition: Cost and frequency deviation minimization

This experiment is aimed at testing the action-composition strategy described in Section 3.3. To this end, for each overarching objective, one set of agents is trained. Set 1, aimed at balancing the system load, is in fact the same model trained in case study I. In Set 2, the state input is composed by the duple $Z_i(t)$, $Z_{\text{total}}(t)$. The model is trained with a single objective reward function aimed at finding the minimum cost of production for every total output as seen below:

$$r_{III-2}(\Delta P_{\text{total}}(t)) = \left(9 \cdot 2^{-\frac{\Delta P^2_{\text{total}}(t)}{2}} + 2^{-\frac{\Delta P^2_{\text{total}}(t)}{100}}\right)\frac{1}{10}. \quad (16)$$

Training was performed by beginning episodes with different output combinations and finding the output set that minimizes the cost of production while keeping the total output level constant. This is done by calculating the individual power levels that minimize the cost for the total output observed at $t = 0$ for every episode, i.e., $p_i^\star(0)$ for each generator i. Those values are then used throughout the episode to calculate the total power deviation ($\Delta P_{\text{total}}(0)$), as shown in (15). The objective of the model trained in Set 2 is then to minimize the sum of the individual generators' output deviation from the cost optimal setup, given a total output. One important point to note is that while Set 2 is trained using a target power that is constant per episode (the total power at the beginning of the episode), during execution the value

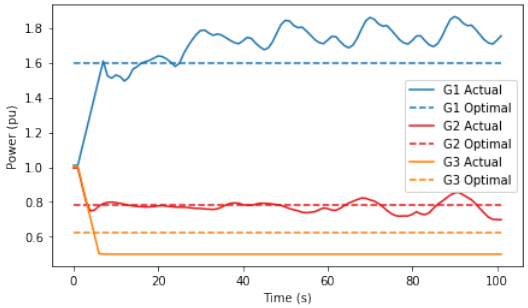

Fig. 8. Case Study III (Cost Model) - Per generator output vs target

used as target power is the current total output observed at every step. This is by design to minimize interference between the actions from both sets. While actions from Set 1 change the total output towards balancing the system, actions from 2 redistribute the total power towards the then optimal output.

The initial results, depicted in Fig. 8, indicate that the cost model is able to perform somewhat as desired. As can be observed, G3 relies on the enforced limits to keep its output constant at the minimum value. G2 show some oscillation but is centered around its desired optimal value. Finally, G1 has an oscillatory pattern similar to G2, but offsets above its desired value. These deviations as oscillatory patterns are too large for use in a production system. However, it should also be noted that, as was the case in case study I, further tuning of the reward function constants and prolonging the training period had an observable effect in mitigating these behaviours.

Even though the individual models' performances can be used to inform the final results, the action composition approach should be ultimately evaluated with respect to the joint performance of combining both models. In that regard, a few tests were performed with different combinations of weights assigned to the frequency and cost models, as shown in Table 3.

Table 3. Action Composition Weights

Setting	$\rho_{frequency}$	ρ_{cost}
Frequency Dominant	0.7	0.3
Cost Dominant	0.3	0.7

In the frequency dominant study the weight of the frequency action is higher as may be seen in Table 3. Initially, one would expect this composition to result in a harmonious balance between both models. However, this is not the case. As observed in case study I, the trained frequency model relies basically on a single generator to provide most of the output and change its output to gradually balance the system. Furthermore, in this particular instance of the trained model, the generator elected for that role was G3, which also has the characteristic of being the least cost-efficient generator among the set. Together, these characteristics result in a clashing behaviour between both models, as can be seen in Figs 9 and 10.

For generators G1 and G2, the frequency model simply acts to reduce the power indefinitely, relying on the enforcement of the minimum floor. The mixing weights are such that the frequency model continuously overrides the

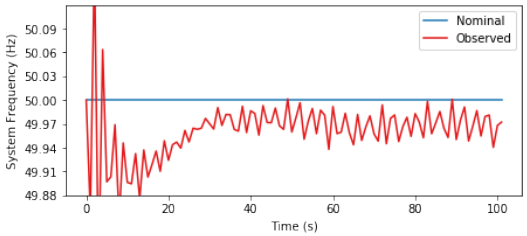

Fig. 9. Case Study III (Frequency Dominant): Observed frequency

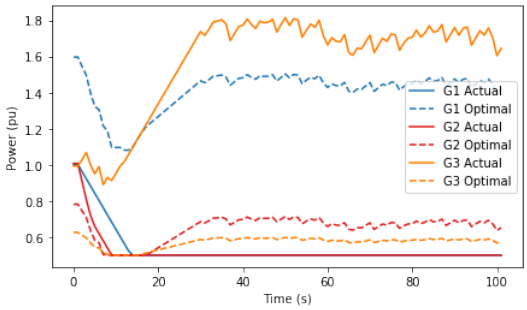

Fig. 10. Case Study III (Frequency Dominant): Generator output vs cost optimal

actions issued by the cost model, resulting in a behaviour much like in the frequency only model discussed in Section 4.2. Generator G3, however, has a uniquely interesting behaviour. Initially, it rises much like in the frequency model. As it approximates the output which would balance the system, the frequency model issues increasingly smaller actions to perform the fine-grained balance of the system. Meanwhile, the cost model continues to issue actions to dramatically lower G3's by virtue of it being the least cost-effective generator and having an output significantly above its optimal value. These divergent actions eventually reach an equilibrium at a point in which the frequency is far enough from the nominal so that the magnitude of the frequency and cost actions are counterbalanced. The final result is a downward shift in the observed frequency. The system displays a steady-state with an oscillatory amplitude similar to the one produced in case study I, but centered around 0.03 Hz below the nominal frequency.

In the cost dominant study, the weight of the cost action is higher as can be seen in Table 3. This test is a mirrored version of the previous one. This change in weights results in the system performing largely as intended, as can be seen in Figs 11 and 12. The final result is such that the system is balanced within 0.03 Hz of the nominal setpoint, while the power output levels approach those that lead to the minimum cost of production. In this case, the downward shift in frequency seen in the frequency dominant test is no longer observed.

5. CONCLUSIONS

In this paper, we formulated the load frequency control problem as a Markov Decision Process and employed reinforcement learning techniques to train autonomous agents able to perform semi-decentralized primary, secondary and tertiary control. We then propose two strategies for dealing with the tradeoffs associated with multiple objectives,

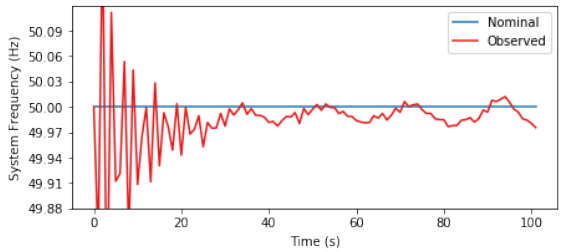

Fig. 11. Case Study III (Cost Dominant): Observed frequency

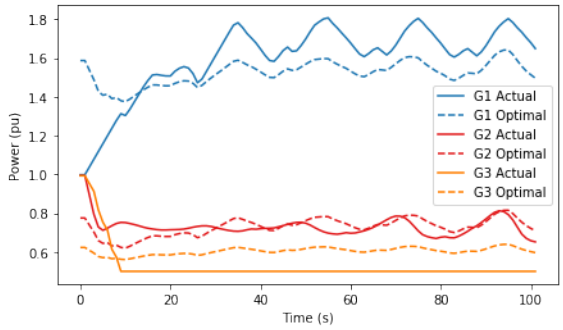

Fig. 12. Case Study III (Cost Dominant): Generator output vs cost optimal

each with its own benefits and disadvantages. Reward Composition consolidates multiple objectives into a single reward function used to train a single set of models, whereas Action Composition trains one set of models per objective and then consolidates the actions issued by all sets. Both methodologies decentralize decision making, but retain some degree of centralization in the form of the total secondary action used in the state input for the models. Overall both approaches were able to restore the system frequency in a cost efficient way, although more work would be required for tuning the solutions, demonstrating its generalizable capabilities and applying it to industrial scenarios. Additionally, the codebase implemented was designed with the explicit intent to allow for further experimentation and expansion.

Future research would include the introduction of more objectives, such as ecological impact of powering the grid, as the methodology employed and codebase developed have no restriction regarding the number of objectives being pursued. Regarding decentralization, one possibility would involve the use of accessory metadata such as timestamps associated with the total secondary action. Intuitively, this could help relax the real-time constraint of the information centralization by enabling agents to rely on stale information for approximating the desired behaviour.

REFERENCES

Ali, A., Khan, B., Mehmood, C.A., Ullah, Z., Ali, S.M., and Ullah, R. (2017). Decentralized mpc based frequency control for smart grid. In *2017 International Conference on Energy Conservation and Efficiency (ICECE)*, 1–6. doi:10.1109/ECE.2017.8248819.

Ambrose, J. (2019). New rules give households right to sell solar power back to energy firms. URL `https://www.theguardian.com/environment/2019/jun/09/energy-firms-buy-electricity-from-household-rooftop-solar-panels`.

Apostolopoulou, D., Sauer, P.W., and Domnguez-Garca, A.D. (2015a). Balancing authority area coordination with limited exchange of information. In *2015 IEEE Power Energy Society General Meeting*, 1–5. doi:10.1109/PESGM.2015.7286133.

Apostolopoulou, D., Sauer, P.W., and Domnguez-Garca, A.D. (2015b). Distributed optimal load frequency control and balancing authority area coordination. In *2015 North American Power Symposium (NAPS)*, 1–5. doi:10.1109/NAPS.2015.7335113.

Heydari, R., Khayat, Y., Naderi, M., Anvari-Moghaddam, A., Dragicevic, T., and Blaabjerg, F. (2019). A decentralized adaptive control method for frequency regulation and power sharing in autonomous microgrids. In *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*, 2427–2432. doi:10.1109/ISIE.2019.8781102.

Kumtepeli, V., Wang, Y., and Tripathi, A. (2016). Multi-area model predictive load frequency control: A decentralized approach. In *2016 Asian Conference on Energy, Power and Transportation Electrification (ACEPT)*, 1–5. doi:10.1109/ACEPT.2016.7811530.

Leggett, T. (2017). How your electric car could be 'a virtual power station'. URL `https://www.bbc.co.uk/news/business-42013625`.

Liu, C., Xu, X., and Hu, D. (2015). Multiobjective reinforcement learning: A comprehensive overview. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(3), 385–398. doi:10.1109/TSMC.2014.2358639.

Lowe, R., Mordatch, I., Abbeel, P., Wu, Y., Tamar, A., and Harb, J. (2017a). Learning to cooperate, compete, and communicate. URL `https://openai.com/blog/learning-to-cooperate-compete-and-communicate/`.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. (2017b). Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 6382–6393. Curran Associates Inc., USA. URL `http://dl.acm.org/citation.cfm?id=3295222.3295385`.

Miller, R. and Malinowski, J. (1994). *Power System Operation*. McGraw-Hill Education.

Moffaert, K.V. and Nowé, A. (2014). Multi-Objective Reinforcement Learning using Sets of Pareto Dominating Policies. Technical report. URL `http://www.jmlr.org/papers/volume15/vanmoffaert14a/vanmoffaert14a.pdf`.

National Grid Electricity Transmission (2017). *The Grid Code*. URL `https://www.nationalgrid.com/sites/default/files/documents/8589935310-Complete%20Grid%20Code.pdf`.

Rozada, S. (2018). Frequency control in unbalanced distribution systems. *City, University of London MSc Data Science Thesis*.

Steitz, C. (2019). Nissan leaf gets approval for vehicle-to-grid use in germany. URL `https://reut.rs/2OISFat`.

Wood, A., Wollenberg, B., and Sheblé, G. (2013). *Power Generation, Operation, and Control*. Wiley.